REMARKS

In response to the First Action Final Rejection dated November 27, 2006, we have tried to schedule an interview to discuss the cited art with the Examiner and also with the Primary Examiner assigned to the case. We will continue trying to establish contact to schedule such an interview, as we believe that the cited Fuoco reference, United States Patent 5,452,429, has little or no teaching that corresponds to the currently pending claims.

As set forth in claim 1, the invention is a method that produces, from stored and regenerated parity check bits, a result that is used to directly **identify (i) the data buffer location that contains a data word with an erroneous bit, and (ii) the position of the erroneous bit in the data word contained in the identified data buffer location.** For clarity, we have changed the format of certain claims to highlight that two distinct things are identified by manipulation of the parity check bits.

We point out that the data words include only data. Further, we point out that the parity check bits are stored separately from the data words. Thus, the current system does not, as is shown in Fuoco, use parity bits that are exclusive to a given data codeword to determine errors within that data codeword. Rather, the current invention utilizes the parity check bits that are associated with data that consists of *multiple* data words, to identify both the buffer location that contains an erroneous data word, and the location within the identified data word of an erroneous bit.

While the Fuoco reference describes an encoding and decoding scheme, the scheme does not produce, from stored and regenerated parity check bits, a result that is usable to directly identify the data buffer location that contains an erroneous data word. Rather, the Fuoco system, like conventional encoding and decoding systems, identifies the location of the erroneous bit within a given data codeword using parity check bits that are exclusively contained in that data codeword. Thus, the Fuoco system does not produce from the stored and regenerated parity check bits a result that is usable to directly identify a data buffer location that contains an erroneous data word that is a part of the data that are encoded to generate the parity check bits, as set forth in pending claim 1.

While the Examiner quotes at length a description of how the data and associated parity check bits are entered into the particular memories used in the Fuoco system, the claims of the current application are not directed to overwriting all or portions of a data codeword contained in memory, rather they are directed to a method of encoding or decoding data. Accordingly, the fact that the Fuoco system uses a particular type of memory with particular lines that are asserted or deasserted to over write all or portions of a stored data codeword is of little consequence, since the encoding and decoding mechanisms used in Fuoco do not teach or suggest the method of encoding or decoding set forth in the pending claims. Specifically, the Fuoco system with its data codewords that consist of data bytes and their corresponding parity check bits does not teach or suggest a method of decoding errors that **produces from the stored and regenerated parity check bits a result that is usable to directly identify (i) the data buffer**

location that contains a data word that has an erroneous bit and (ii) the position of the erroneous bit in the data word contained in the identified data buffer location. Accordingly, the Fuoco reference does not make obvious the invention as set forth in independent claims 1, 24 and 26 and the claims that depend therefrom.

We again point out that the invention of claim 23 is a particular method of encoding, not a method of overwriting data. The encoding method includes the use of a generator matrix that has a parity check generation portion that comprises rows of bits that correspond to binary representations of the respective buffer locations that are used to store the data words. There is no teaching or suggestion of such an encoding method step in the Fuoco reference. Again, the mechanism by which Fuoco overwrites all or portions of a particular data codeword that consists of data bytes and their corresponding parity check bits contained in a particular memory location has little to do with the encoding method that is the subject of claim 23. For example, the Fuoco system does not supply *multiple* data words to a generator matrix to produce the parity check bits.

The system claim of 25 similarly includes the generator matrix that is discussed above with respect to claim 23. The Fuoco reference does not teach such a generator matrix, but instead teaches its own particular syndrome generation mechanism that differs considerably from the mechanism used in the current system. The only commonality between the two systems is that each uses a generator matrix. However, the generator matrices of the two systems are completely different, as are the objectives of the encoding schemes of the current invention and the Fuoco reference.

In particular, the current system uses an encoding mechanism that allows the buffer location of an erroneous data word to be determined using parity check bits that are associated with the data as a whole, not just the data word. In contrast, the Fuoco encoding mechanism is designed such that particular data and its associated parity check bits that together form a data codeword can be completely overwritten or partially overwritten depending on whether or not every byte of the data in the data codeword are to be replaced or only selected bytes of the data in the data codeword are to be replaced. Accordingly, the encoding mechanism of Fuoco and/or how Fuoco writes data to the location that contains a particular data codeword by asserting particular lines and so forth do not teach or suggest the system of claim 23. Specifically, Fuoco does not teach or suggest a controller that is operable to apply data to a generator matrix comprising a data portion and a parity check generation portion, with the parity check generation portion comprising **rows of bits corresponding to binary representations of the respective data buffer locations to be used to store the data.** For the same reasons, the apparatus of claims 26 and 27 are also not taught or suggested by the Fuoco system because, *inter alia*, the Fuoco reference does not teach or suggest the use of the parity generator matrix set forth in those claims.

Further, there is no teaching or suggestion in Fuoco of the invention as set forth in dependent claims 8 and 11. Fuoco does not teach or suggest producing a parity check bit that corresponds to the parity check bits generated for the data. Fuoco teaches generating parity check bits only for the data bytes in a given data code word - see Table 1.

We do not specifically address the Examiner's rejections of certain of the dependent claims. This should not be construed as acquiescence to the rejections, but as recognition that the rejections are moot based on our remarks regarding the allowability of the independent claims.

We respectfully request that the Examiner contact the undersigned to hold an interview to discuss the Fuoco reference and the current invention in more detail before responding to this Amendment.

We look forward to discussing the cited art with the Examiner. Further, we point out that the claims, as amended for clarity, should now be in form for allowance, and we request that the amendments be entered and a Notice of Allowance be issued for all pending claims.

Please charge any additional fee occasioned by this paper to our Deposit Account No. 03-1237.

Respectfully submitted,

Patricia A. Sheehan
Reg. No. 32,301
CESARI AND MCKENNA, LLP
88 Black Falcon Avenue
Boston, MA 02210-2414
(617) 951-2500